VIIRS Active Fires Science Processing Algorithm (VIIRS-AF_SPA) User's Guide

Version 1.3.1

July 2014

GODDARD SPACE FLIGHT CENTER
GREENBELT, MARYLAND

VIIRS Active Fires Science Processing Algorithm

VIIRS-AF_SPA

General

The NASA Goddard Space Flight Center's (GSFC) Direct Readout Laboratory (DRL), Code 606.3 developed this software for the International Polar Orbiter Processing Package (IPOPP). The IPOPP package maximizes the utility of Earth science data for making real-time decisions by giving fast access to instrument data and derivative products from the Suomi National Polar-orbiting Partnership (SNPP), Aqua, and Terra missions and, in the future, the Joint Polar Satellite System (JPSS) mission.

Users must agree to all terms and conditions in the Software Usage Agreement on the DRL Web Portal before downloading this software.

Software and documentation published on the DRL Web Portal may occasionally be updated or modified. The most current versions of DRL software are available at the DRL Web Portal:

http://directreadout.sci.gsfc.nasa.gov/?id=software

Questions relating to the contents or status of this software and its documentation should be addressed to the DRL via the Contact DRL mechanism at the DRL Web Portal:

http://directreadout.sci.gsfc.nasa.gov/?id=dspContent&cid=66

Algorithm Wrapper Concept

The DRL has developed an algorithm wrapper to provide a common command and execution interface to encapsulate multi-discipline, multi-mission science processing algorithms. The wrapper also provides a structured, standardized technique for packaging new or updated algorithms with minimal effort.

A Science Processing Algorithm (SPA) is defined as a wrapper and its contained algorithm. SPAs will function in a standalone, cross-platform environment to serve the needs of the broad Direct Readout community. Detailed information about SPAs and other DRL technologies is available at the DRL Web Portal.

Software Description

This software package contains the Visible Infrared Radiometer Suite Active Fires Science Processing Algorithm (VIIRS-AF_SPA). This algorithm primarily uses brightness temperatures derived from bands M13 and M15 to detect fires. VIIRS bands M5, M7, M11 and M16 are used to reject false alarms and to mask clouds. The algorithm takes as input VIIRS Science Data Record (SDR) files, along with the associated geolocation file, and identifies active fires. The outputs are a two-dimensional fire mask in Hierarchical Data Format (HDF) and a fire location list text file. The SPA functions in two modes: Standalone, or as an IPOPP plug-in.

Software Version

Version 1.5 of the DRL algorithm wrapper was used to package the SPA described in this document. The SPA uses version 1.3.1 of the VIIRS-AF algorithm.

Enhancements to this SPA include:

- Update of the algorithm to version 1.3.1. This version of the algorithm has been synchronized with its heritage MODIS counterpart (i.e., MOD14) to fulfill NASA's objective of measurement continuity. VIIRS-AF_SPA v1.3.1 includes all relevant changes made to the MOD14 algorithm as part of the formal MODIS algorithm evolution (presently Collection 6 series) process.
- Capability to accept compressed HDF5 input files.

This software will execute on a 64-bit computer and has been tested on computers with 32GB of RAM, with the following operating systems:

a) Fedora 18 X86_64;

b) CentOS Linux 6.4 X86_64;

c) OpenSUSE Linux 12.1 X86_64;

d) Kubuntu 13.04 X86_64.

Credits

The VIIRS-AF algorithm was co-developed by the Land Science Team and the DRL at NASA/GSFC, under the guidance of the International Land Direct Readout Coordinating Committee (ILDRCC).

Prerequisites

To run this package, you must have the Java Development Kit (JDK) or Java Runtime Engine (JRE) (Java 1.6.0_25 or higher) installed on your computer, and have the Java installation bin/ subdirectory in your PATH environment variable. This package contains 64-bit binaries statically pre-compiled on an x86-compatible 64-bit computer running under Fedora 14, using gcc 4.5.1.

Program Inputs and Outputs

See the Program Operation subsection.

Installation and Configuration

Installing as a Standalone Application: Download the VIIRS-AF_1.3.1_SPA_1.5.tar.gz and VIIRS-AF_1.3.1_SPA_1.5_testdata.tar.gz (optional) files into the same directory.

Decompress and un-archive the VIIRS-AF_1.3.1_SPA_1.5.tar.gz and
VIIRS-AF_1.3.1_SPA_1.5_testdata.tar.gz (optional) files:

```
$ tar –xzf VIIRS-AF_1.3.1_SPA_1.5.tar.gz
$ tar –xzf VIIRS-AF_1.3.1_SPA_1.5_testdata.tar.gz
```

This will create the following subdirectories:

```
SPA
    VIIRS-AF
        algorithm
        ancillary
        station
        wrapper
        testscripts
        testdata
```

Installing into an IPOPP Framework: This SPA can also be installed dynamically into an IPOPP framework to automate production of VIIRS-AF data products. The SPA installation process will install SPA station(s) into IPOPP. An SPA station is an IPOPP agent that provides the mechanism necessary for running an SPA automatically within the IPOPP framework. Once this SPA is installed, users must enable the station(s) corresponding to this SPA along with any other pre-requisite station(s). Instructions for installing an SPA and enabling its stations are contained in the IPOPP User’s Guide (available on the DRL Web Portal along with the IPOPP package). The SPA stations associated with this SPA are listed in Appendix A.

Software Package Testing and Validation

The testscripts subdirectory contains test scripts that can be used to verify that your current installation of the SPA is working properly, as described below. Note that the optional VIIRS-AF_1.3.1_SPA_1.5_testdata.tar.gz file is required to execute these testing procedures.

Step 1: cd into the testscripts directory.

Step 2: Run the 'run-vaf.sh' script by typing:

```
$./run-vaf.sh
```

A successful execution usually takes some time (a few seconds, depending on the speed of your computer), so if the execution seems to get stuck, do not become impatient. If everything is working properly, the script will terminate with a message such as:

```
Output viirs.ActiveFires is /home/ipopp/SPA/VIIRS-AF/testdata/output/NPP_VIIRS-AF.hdf
Output viirs.FireLoc is /home/ipopp/SPA/VIIRS-AF /testdata/output/FireLoc.txt
```

You can cd to the output directory to verify that the science product(s) exist. If there is a problem and the code terminates abnormally, the problem can be identified using the log files. Log files are generated automatically within the directory used for execution. They start with stdfile* and errfile*. Other log and intermediate files may be generated automatically within the directory used for execution. They are useful for traceability and debugging purposes. However it is strongly recommended that users clean up log files and intermediate files left behind in the run directory before initiating a fresh execution of the SPA. Intermediate files from a previous run may affect a successive run and produce ambiguous results. Other problems may be caused by incompatibility between your system and the binaries provided with this software package. Please report any errors that cannot be fixed to the DRL.

Test output product(s) are available for comparison in the testdata/output directory. These test output product(s) were generated on a 64-bit PC architecture computer running Fedora 14. The output products serve as an indicator of expected program output. Use a comparison utility (such as diff, hdiff, etc.) to compare your output product(s) to those provided in the testdata/output directory. Locally generated files may differ slightly from the provided output files because of differences in machine architecture or operating systems.

Program Operation

In order to run the package using your own input data, you can either use the run scripts within the wrapper directory, or modify the test script within the testscripts directory.

To Use the Run Scripts

Identify the 'run' scripts: The wrapper directory within this package contains a subdirectory named VIIRS-AF which contains an executable called 'run'. You must execute the 'run' within the wrapper subdirectory to execute the process. Note that to execute 'run', you need to have java on your path.

Specify input parameters using <label value> pairs: To execute the 'run' scripts, you must supply the required input and output parameters. Input and output parameters are usually file paths or other values (e.g., the satellite name). Each parameter is specified on the command line by a <label value> pair. Labels are simply predefined names for parameters. Each label must be followed by its actual value. Each process has its own set of <label value> pairs that must be specified in order for it to execute. Some of these pairs are optional, meaning the process would still be able to execute even if that parameter is not supplied. There are two types of <label value> pairs that the VIIRS Active Fires SPA uses, as follows:

a) Input file labels/values. These are input file paths. Values are absolute or relative paths to the corresponding input file.

b) Output file labels. These are output files that are produced by the SPA. Values are the relative/absolute paths of the files you want to generate.

The following table contains labels, and their descriptions, required by the VIIRS-AF_SPA.

Input File Labels	Description	Source
viirs.gmtco	VIIRS SDR GMTCO level 1 file	1. The C-SDR_SPA and VIIRS-SDR SPAs can be used to create these VIIRS SDR products. 2. Real time VIIRS SDR products over the eastern US region are available from the DRL ftp site at: ftp://is.sci.gsfc.nasa.gov/gsfcdata/npp/viirs/level1/<SVMxx|GMTCO>_npp_dyyyymmdd_thhmmssS_ehhmmssS*.h5 Where yyyy, mm, dd represents the year, month, and date for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath. 3. VIIRS SDR products for other locations and times are available for download at www.class.noaa.gov
viirs.svm05	VIIRS SDR SVM05 level 1 file	
viirs.svm07	VIIRS SDR SVM07 level 1 file	
viirs.svm11	VIIRS SDR SVM11 level 1 file	
viirs.svm13	VIIRS SDR SVM13 level 1 file	
viirs.svm15	VIIRS SDR SVM15 level 1 file	
viirs.svm16	VIIRS SDR SVM16 level 1 file	

Output File Labels	Description	Destination (when SPA is installed in IPOPP)
viirs.ActiveFires	VIIRS Active Fires HDF file (VAF)	/raid/pub/gsfcdata/npp/viirs/level2/VAF_npp_dyyyymmdd_thhmmssS_ehhmmssS*.hdf Where yyyy, mm, dd represents the year, month, and date for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath.
viirs.FireLoc (optional)	VIIRS Fire-location text file	/raid/pub/gsfcdata/npp/viirs/level2/FireLoc_npp_dyyyymmdd_thhmmssS_ehhmmssS*.txt Where yyyy, mm, dd represents the year, month, and date for the start of the swath; the first hh, mm, ss, S represents the hour, minutes, seconds, and 10th of a second for the start of the swath and the second hh, mm, ss, S represents the end time of the swath.

NOTE: The Fire Location text file contains information about each fire pixel detected by the SPA. Each line represents one fire pixel and has the following seven columns, separated by commas.

Column 1: Fire Pixel Latitude

Column 2: Fire Pixel Longitude

Column 3: M13 Brightness Temperature

Column 4: Along scan pixel dimension (km)

Column 5: Along track pixel dimension (km)

Column 6: Fire Detection Confidence

Column 7: Fire Radiative Power (FRP)

Execute the 'run': The following is an example of a command line to run the VIIRS-AF process from the testscripts subdirectory. You can run it from the directory of your choice by using the correct path to the 'run' script and your datasets.

```
$ ../wrapper/VIIRS-AF/run \

viirs.gmtco \
../testdata/input/GMTCO_npp_d20130323_t1851552_e1853194_b07270_c20130329144438416689_noaa_ops.h5 \
viirs.svm05 \
../testdata/input/SVM05_npp_d20130323_t1851552_e1853194_b07270_c20130329144411503651_noaa_ops.h5 \
viirs.svm07 \
 ../testdata/input/SVM07_npp_d20130323_t1851552_e1853194_b07270_c20130329144352689405_noaa_ops.h5 \
viirs.svm11 \
 ../testdata/input/SVM11_npp_d20130323_t1851552_e1853194_b07270_c20130329144429442736_noaa_ops.h5 \
viirs.svm13 \
 ../testdata/input/SVM13_npp_d20130323_t1851552_e1853194_b07270_c20130329144529590889_noaa_ops.h5 \
viirs.svm15 \
 ../testdata/input/SVM15_npp_d20130323_t1851552_e1853194_b07270_c20130329144411111248_noaa_ops.h5 \
viirs.svm16 \
 ../testdata/input/SVM16_npp_d20130323_t1851552_e1853194_b07270_c20130329144517993558_noaa_ops.h5 \
viirs.ActiveFires ../testdata/output/NPP_VIIRS-AF.hdf \
viirs.FireLoc ../testdata/output/FireLoc.txt
```

```
Output viirs.ActiveFires is /home/ipopp/SPA/VIIRS-AF/testdata/output/NPP_VIIRS-AF.hdf
Output viirs.FireLoc is /home/ipopp/SPA/VIIRS-AF /testdata/output/FireLoc.txt
```

A successful execution usually requires a few seconds, depending on the speed of your computer and the number of granules in the input files. If execution fails, you will see an error message indicating the cause of failure (e.g., a file cannot be found, or a label cannot be recognized). Correct the problem and run again. If the problem has some other cause, it can be identified using the log files. Log files are automatically generated within the directory used for execution. They start with stdfile* and errfile* and can be deleted after execution. Other log and intermediate files may be generated automatically within the directory used for execution. They are useful for traceability and debugging purposes. However it is strongly recommended that users clean up log files and intermediate files left behind in the run directory before initiating a fresh execution of the SPA. Intermediate files from a previous run may affect a successive run and produce ambiguous results. Other problems may be caused by incompatibility between your system and the binaries provided with this software package.

To Use the Script in the testscripts Directory

One simple way to run the algorithm from any directory of your choice, using your own data, is to copy the run-vaf.sh script from the testscripts directory to the selected directory. Change the values of the different variables to reflect the file paths of the wrapper directories and the input/output files. Then modify the input/output file names. Run the script to process your data.

Appendix A
SPA Stations

Installation of this SPA in IPOPP mode will make the SPA stations listed in Table A-1 available to IPOPP. These stations along with any other pre-requisite stations (listed in Table A-2) will need to be enabled to allow IPOPP to automate production of the VIIRS-AF data products. Further, users who wish to generate image products from the data products generated by this SPA will need to enable the image-generating stations listed in Table A-3. The SPAs containing the pre-requisite and the image-generating stations listed in Tables A-2 and A-3 can be downloaded from the DRL Web Portal, in case they are not already available in your IPOPP installation. Details about these other SPAs are available in the respective SPA User's Guides. Please refer to the IPOPP User's Guide for instructions on how to install an SPA in IPOPP and enable the corresponding stations.

Table A-1. SPA Stations

SPA stations for this SPA	Data Products produced
VIIRS-AF	VIIRS Active Fires HDF file VIIRS Fire-location text file

Table A-2. Pre-requisite Stations

Pre-requisite SPA stations	SPA in which they are available
VIIRS_C-SDR	C-SDR_SPA
OR	
VIIRS-SDR	VIIRS-SDR_SPA

WARNING: The stations VIIRS-SDR and VIIRS_C-SDR must never be run simultaneously.

Table A-3. Image-generating Stations

Image-generating stations	SPA in which they are available
viirsaf-geotiff	H2G_SPA
vcviirsfire-geotiff	H2G_SPA

NOTE: The vcviirsfire-geotiff station additionally needs CVIIRS_SPA to be installed and enabled in IPOPP in order to run. CVIIRS_SPA is available for download from the DRL Web Portal in case it is not already available in your IPOPP installation.